

A Rao-Blackwellized Parts-Constellation Tracker

Grant Schindler and Frank Dellaert

College of Computing, Georgia Institute of Technology
{schindler, dellaert}@cc.gatech.edu

Abstract

We present a method for efficiently tracking objects represented as constellations of parts by integrating out the shape of the model. Parts-based models have been successfully applied to object recognition and tracking. However, the high dimensionality of such models present an obstacle to traditional particle filtering approaches. We can efficiently use parts-based models in a particle filter by applying Rao-Blackwellization to integrate out continuous parameters such as shape. This allows us to maintain multiple hypotheses for the pose of an object without the need to sample in the high-dimensional spaces in which parts-based models live. We present experimental results for a challenging biological tracking task.

1. Introduction

We are interested in tracking insects in video, a task complicated by the fact that insects exhibit non-rigid motion. Like other tracking targets, such as people, insects are physically composed of multiple parts that flex and bend with respect to each other. We would like to model this flexible motion, which is hypothesized to improve the performance of our tracker and increase the richness of the data that can be used for subsequent analysis. As such, we adopt a model that incorporates an object's individual parts, modeling the joint configuration of the parts as a whole, and modeling the appearance of each part individually. We show how to efficiently incorporate such a model into a particle filter by treating the shape analytically and sampling only over pose, a process commonly known as *Rao-Blackwellization*. We use Expectation-Maximization (EM) to learn appearance and shape parameters for these models and perform tracking with a Rao-Blackwellized particle filter.

We adopt the framework of [5] to model insects as flexible constellations of parts. Though parts-based models have a long history of use in computer vision, a powerful probabilistic formulation for modeling objects composed of flexible parts was first offered by Burl, Weber, and Perona [2] and later extended by Fergus, Perona, and Zisserman [5]. In their formulation, each part has a location, appearance, and relative scale, and the shape of an object is represented as



Figure 1: *Parts-constellation model of a bee.* We learn a joint shape distribution on part configurations, as well as an appearance model for each part. The mean appearance and pose of each part are shown above. Ellipses show individual part covariances. By integrating over shape, we can efficiently incorporate such a model into a particle filter.

the relative location of its parts. We apply this framework to the problem of tracking moving objects in video. Other parts-based methods have been used for tracking as well. A parts-based method for tracking loose-limbed people in 3D over multiple views is presented in [15], which makes use of bottom-up part-detectors to detect possible part locations in each frame. Our method takes a related approach, but uses an image registration technique based on the well-known Lucas-Kanade algorithm [10] for locally registering part templates. In contrast to [15], we are tracking a target across a single view containing many other identical targets.

Rao-Blackwellization, as applied to particle filters, is a method to analytically compute a portion of the distribution over the state space, so as to reduce the dimensionality of the sampled space and the number of samples used. Rao-Blackwellized particle filters (RBPFs) have previously been applied to several estimation problems, including insect tracking. In [9], an RBPF was used to incorporate subspace appearance models into particle filter tracking of insects. In [14], the authors integrate over the 2D target positions and sample over measurement target assignments to track people based on noisy position measurements from IR sensors. In [6], de Freitas uses an RBPF for fault detection where Kalman filters are applied over continuous parameters and samples obtained over discrete fault states. And finally, in [12], the authors integrate over landmark locations in a robotics application where the goal is to localize a robot while simultaneously building a map of the environment.

2. A Bayesian Filtering Approach

Bayesian filtering is a traditional approach to the target tracking problem in which, at time t , we recursively estimate the posterior distribution $P(X_t|Z_{1:t})$ of some state X_t conditioned on all measurements $Z_{1:t}$ up to time t as:

$$P(X_t|Z_{1:t}) \propto P(Z_t|X_t) \int_{X_{t-1}} P(X_t|X_{t-1})P(X_{t-1}|Z_{1:t-1}) \quad (1)$$

We call $P(Z_t|X_t)$ the *measurement model* and $P(X_t|X_{t-1})$ the *motion model*.

When applying a Bayes filter to the problem of parts-based tracking, the state $X_t = (Y_t, S_t)$ has two components: the *pose* Y_t of the target and the *shape* S_t describing the configuration of parts. The measurements $Z_{1:t} = I_{1:t}$ are images I_t observed at time t in a video sequence.

By analogy to equation (1), if we wanted to compute the posterior distribution $P(Y_t, S_t|I_{1:t})$ on both pose Y_t and shape S_t , the corresponding Bayes filter would be computed by integrating over both the pose Y_{t-1} and the shape S_{t-1} at the previous time step $t - 1$:

$$P(Y_t, S_t|I_{1:t}) = kP(I_t|Y_t, S_t) \int_{Y_{t-1}} \int_{S_{t-1}} P(Y_t, S_t|Y_{t-1}, S_{t-1})P(Y_{t-1}, S_{t-1}|I_{1:t-1}) \quad (2)$$

By integrating over the current shape S_t on both sides of equation 2 we obtain a marginal filter on the pose Y_t alone :

$$P(Y_t|I_{1:t}) = k \int_{S_t} P(I_t|Y_t, S_t) \times \int_{Y_{t-1}} \int_{S_{t-1}} P(Y_t, S_t|Y_{t-1}, S_{t-1})P(Y_{t-1}, S_{t-1}|I_{1:t-1})$$

In our model, we will assume that (a) the motion of the target is independent of shape, and (b) that there is no temporal coherence to the shape. Taking into account these independence assumptions the joint motion term factors into the product of a simpler motion model $P(Y_t|Y_{t-1})$ and a shape model $P(S_t|Y_t)$:

$$P(Y_t, S_t|Y_{t-1}, S_{t-1}) \propto P(Y_t|Y_{t-1})P(S_t|Y_t)$$

Thus the final form of our exact marginal Bayes filter is:

$$P(Y_t|I_{1:t}) = k \int_{S_t} P(I_t|Y_t, S_t)P(S_t|Y_t) \times \int_{Y_{t-1}} \int_{S_{t-1}} P(Y_t|Y_{t-1})P(Y_{t-1}, S_{t-1}|I_{1:t-1}) \quad (3)$$

We describe a Monte Carlo approximation of this Bayes filtering distribution in Section 4.

3. The Parts-Constellation Model

To fully specify the above Bayes filter in equation (3), we need to define an appearance model $P(I_t|Y_t, S_t)$, a shape model $P(S_t|Y_t)$, and a motion model $P(Y_t|Y_{t-1})$. Here, we describe our appearance and shape models in more detail. The motion model does not depend on shape, and is thus not specific to our approach.

3.1. Appearance Model

If we define the image I as the union of foreground and background image regions $F(Y, S)$ and $B(Y, S)$, whose position and extent have a functional dependence on both pose Y and shape S , the appearance model factors as:

$$\begin{aligned} P(I|Y, S) &= P(F(Y, S), B(Y, S)|Y, S) \\ &= P_F(F(Y, S))P_B(B(Y, S)) \end{aligned}$$

Here P_F and P_B are distributions on the foreground and background models, respectively. This factorization is valid if we assume no overlap between foreground and background regions in the image. Similar to the approach taken by [2] and [5], we can arrive at a formulation of the image likelihood purely in terms of F , the foreground region of interest, by multiplying both sides of this expression by a constant:

$$\begin{aligned} &= P_F(F(Y, S))P_B(B(Y, S)) \frac{P_B(F(Y, S))}{P_B(F(Y, S))} \\ &= P_B(F(Y, S), B(Y, S)) \frac{P_F(F(Y, S))}{P_B(F(Y, S))} \\ &\propto \frac{P_F(F(Y, S))}{P_B(F(Y, S))} \end{aligned}$$

Finally, we break the foreground F into multiple regions F_n corresponding to the individual parts of the model, obtaining a product of likelihood ratios

$$P(I|Y, S) \propto \prod_n \frac{P_{F_n}(F_n(Y, S))}{P_B(F_n(Y, S))} \quad (4)$$

where each part of the foreground F_n is evaluated according to a different foreground distribution P_{F_n} .

3.2. Shape Model

Shape is modeled as a joint Gaussian distribution $P(S|Y)$ on part positions and is parameterized by $\theta_{shape} = \{\mu_{shape}, \Sigma_{shape}\}$. For example, if there are N parts and each part has both a location and an orientation, then Σ_{shape} is a full $3N \times 3N$ covariance matrix. This is similar to the shape model from [5]. The shape model is conditioned on pose Y simply because the mean μ_{shape} is defined with respect to the target's current position.

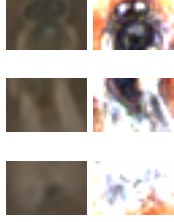


Figure 2: Learned parameters for the foreground image models of the parts of a bee. The left column shows the mean pixel values of each part, while the right column shows the pixel variance.

4. A Rao-Blackwellized Particle Filter

Our Bayes filtering distribution can be approximated with a particle filter [7, 8, 3] in which the posterior $P(Y_{t-1}, S_{t-1} | I_{1:t-1})$ is represented by a set of weighted particles. Using a traditional particle filter, we would sample a pose Y_t from the motion model $P(Y_t | Y_{t-1})$, sample a shape S_t from the shape model $P(S_t | Y_t)$, and then weight this joint sample using the appearance model $P(I_t | Y_t, S_t)$. However, these joint samples on pose and shape live in such a high-dimensional space that approximating the posterior distribution requires an intractably large number of particles, potentially making a parts-based model infeasible.

In a ‘‘Rao-Blackwellized’’ particle filter (RBPF) [13] we analytically integrate out shape and only sample over pose. Thus, we can achieve the same performance as a traditional particle filter with vastly fewer particles. As in [9], we approximate the posterior $P(Y_t, S_t | I_{1:t})$ over the state $\{Y_t, S_t\}$ with a set of hybrid particles $\{Y_t^{(j)}, w_t^{(j)}, \alpha_t^{(j)}(S_t)\}_{j=1}^M$, where $w_t^{(j)}$ is the particle’s importance weight and each particle has its own conditional distribution $\alpha_t^{(j)}(S_t)$ on shape S_t :

$$\begin{aligned} \alpha_t^{(j)}(S_t) &\triangleq P(S_t | Y_t^{(j)}, I_{1:t}) \\ &\propto P(I_t | Y_t^{(j)}, S_t) P(S_t | Y_t^{(j)}) \end{aligned}$$

The hybrid samples constitute a Monte Carlo approximation to the posterior term $P(Y_t, S_t | I_{1:t})$ as follows:

$$\begin{aligned} P(Y_t, S_t | I_{1:t}) &\approx P(Y_t | I_{1:t}) P(S_t | Y_t, I_{1:t}) \quad (5) \\ &\approx \sum_j w_t^{(j)} \delta(Y_t^{(j)}, Y_t) \alpha_t^{(j)}(S_t) \end{aligned}$$

By substituting this approximate posterior (5) into formula (3) for the exact marginal Bayes filter, we obtain the follow-

ing Monte Carlo approximation to the marginal filter:

$$\begin{aligned} P(Y_t | I_{1:t}) &\approx k \int_{S_t} P(I_t | Y_t, S_t) P(S_t | Y_t) \\ &\quad \times \sum_i w_{t-1}^{(i)} P(Y_t | Y_{t-1}^{(i)}) \alpha_{t-1}^{(i)}(S_{t-1}) \end{aligned}$$

Thus, the Rao-Blackwellized particle filter proceeds through the following steps:

Algorithm 1 Rao-Blackwellized Parts-Constellation Filter

1. Select a particle $Y_{t-1}^{(i)}$ from the previous time step according to weights $w_{t-1}^{(i)}$.

2. Sample a new particle $\hat{Y}_t^{(j)}$ from the motion model

$$P(Y_t | Y_{t-1}^{(i)})$$

3. Calculate the posterior density $\alpha_t^{(j)}(S_t)$ on shape S_t :

$$\alpha_t^{(j)}(S_t) = P(I_t | Y_t^{(j)}, S_t) P(S_t | Y_t^{(j)})$$

4. Compute the importance weight $w_t^{(j)} = \int_{S_t} \alpha_t^{(j)}(S_t)$ (see Section 4.1 below).
-

4.1. Computing Importance Weights

The importance weight computation involves an integration over shape S_t , but it is tractable because $\alpha_t^{(j)}(S_t)$ is a Gaussian. The integral of *any* function $q(x) \triangleq k \exp\{-\frac{1}{2}\|\mu - x\|_\Sigma^2\}$ proportional to a Gaussian is

$$\int_x q(x) = k \int_x \exp\left\{-\frac{1}{2}\|\mu - x\|_\Sigma^2\right\} = k \sqrt{|2\pi\Sigma|}$$

where $\|\mu - x\|_\Sigma^2 \triangleq (\mu - x)^T \Sigma^{-1} (\mu - x)$ is the squared Mahalanobis distance from x to μ with covariance matrix Σ . Note that the constant k is equal to $q(\mu)$, as

$$q(\mu) = k \exp\left\{-\frac{1}{2}\|\mu - \mu\|_\Sigma^2\right\} = k \exp\left\{-\frac{1}{2}0\right\} = k$$

Hence, with μ the mode of $q(x)$, we have

$$\int_x q(x) = q(\mu) \sqrt{|2\pi\Sigma|} \quad (6)$$

Observe that if $\alpha_t^{(j)}(S_t)$ is a product of all Gaussian terms, then it is itself Gaussian. Thus, the only constraint on our model is that the shape model, foreground $P_{F_n}(\cdot)$ and background $P_B(\cdot)$ models be normally distributed.

We find the mode of $\alpha_t^{(j)}(S_t)$, which we denote by S_t^* , by optimization. We use an inverse-compositional variant of the Lucas-Kanade algorithm [10, 1] which optimizes the shape by registering templates for each part (the means of the foreground distributions $P_{F_n}(\cdot)$) to the measured image I_t . See section 6 for an explanation of the assumptions underlying this approach. Finally, we apply the above property of Gaussians (6) to compute our importance weight:

$$\begin{aligned} w_t^{(j)} &= \int_{S_t} \alpha_t^{(j)}(S_t) \\ &= \alpha_t^{(j)}(S_t^*) \sqrt{|2\pi\Sigma|} \\ &= P(I_t|Y_t^{(j)}, S_t^*) P(S_t^*|Y_t^{(j)}) \sqrt{|2\pi\Sigma|} \\ &\propto \prod_n \frac{P_{F_n}(F_n(Y_t^{(j)}, S_t^*))}{P_B(F_n(Y_t^{(j)}, S_t^*))} P(S_t^*|Y_t^{(j)}) \sqrt{|2\pi\Sigma|} \end{aligned}$$

5. Learning Model Parameters

For a part-based model with N parts, we must learn the shape parameters $\theta_S = \{\mu_S, \Sigma_S\}$, and appearance parameters $\theta_I = \{\theta_{F_1}, \dots, \theta_{F_N}, \theta_B\}$ (see figure 2). Given a set of training images $I_{1:T}$, we use expectation-maximization (EM) [4, 11], starting from an initial estimate for the parameters $\theta = \{\theta_S, \theta_I\}$. Here, we assume pose Y is given and treat shape S as a hidden variable.

E-Step The goal of the E-step is to calculate the posterior distribution $P(S|Y, I, \theta)$ on shape for each training image given the current estimate of the parameters θ . Note that this distribution is almost identical to the conditional distribution $\alpha(S)$ on shape from the RBPF. Thus, in the E-step, we essentially create a set of hybrid particles each with given pose Y_t and distribution $\alpha_t(S_t|\theta)$ defined with respect to the current parameter estimates:

$$P(S|Y, I, \theta) \propto \prod_n \frac{P_{F_n}(F_n(Y, S)|\theta_{F_n})}{P_B(F_n(Y, S)|\theta_B)} P(S|Y, \theta_S) \quad (7)$$

M-Step In the M-step, we maximize the expected log-posterior $Q(\theta; \theta_{old})$ with respect to θ to obtain a new set of parameters θ_{new} .

$$\theta_{new} = \underset{\theta}{\operatorname{argmax}} \quad Q(\theta; \theta_{old})$$

We define S_t^* as the optimal shape for training image I_t according to equation 7. This optimal shape is obtained using image registration as explained before in Section 4.1. We compute the expected log-posterior $Q(\theta; \theta_{old})$:

$$\begin{aligned} &= E[\log P(I|Y, S, \theta_I) + \log P(S|Y, \theta_S)]_{P(S|Y, I, \theta_{old})} \\ &= \sum_t \left[\sum_n [\log P_{F_n}(F_n(Y_t, S_t^*)|\theta_{F_n}) - \right. \\ &\quad \left. \log P_B(F_n(Y_t, S_t^*)|\theta_B)] + \log P(S_t^*|Y_t, \theta_S) \right] \end{aligned}$$

Intuitively, after we find the set of optimal shapes $S_{1:T}^*$, finding the θ that maximizes $Q(\theta; \theta_{old})$ is equivalent to calculating the mean and covariance directly from the shapes and appearances defined by $S_{1:T}^*$. Note that the background parameters θ_B are not updated during EM.

6. Experimental Results



Figure 3: Tracking a dancing honey bee in an active hive is a difficult task, further complicated by the non-rigid shape of the bee and the complex “waggle” portion of the dance.

We used the parts-based Rao-Blackwellized particle filter to track a dancing honey bee in an active hive (see figure 3), a difficult task that is of interest to biologists studying honey bee behavior. There are over 120 bees in frame throughout the video and they often collide and/or occlude each other. The “waggle” portion of a bee’s dance involves complex motion that is not easily modeled, while the “turning” portion of the dance bends a bee’s body in a way that is difficult to model with only a single template.

We represent the pose of a bee as a vector $Y = \{x, y, \theta\}$ including 2D position (x, y) and rotation θ . The center of rotation of each target is 20 pixels from the front of its head. We model a bee as consisting of $N = 3$ parts of size 30x20 pixels each. Each part has its own pose $\{x_n, y_n, \theta_n\}$ with respect to the pose Y of the entire bee, such that the shape S of bee composed of N parts is represented by a $3 \times N$ dimensional vector. Therefore, for our task, the shape model is a 9-dimensional Gaussian on joint shape configuration S , the motion model is a 3-dimensional Gaussian on change in pose Y , and the appearance model puts a 3-dimensional (R,G,B) Gaussian on every pixel of each foreground region $F_{1:N}$, while the background B is modeled with a single 3-dimensional Gaussian on the average color of a region.

The parameters of the shape, appearance, and motion models were learned from a training data set consisting of

Tracking Method	Particles	Failures	Mean Translation Error	Mean Rotation Error	Mean Time/Frame
Lucas-Kanade	-	49	4.87 pixels	0.42 rad	0.25 s
1-Part PF	540	50	2.63 pixels	0.12 rad	0.89 s
3-Part RBPF	80	24	5.97 pixels	0.27 rad	0.85 s

Table 1: The parts-based RBPF using only 80 particles fails less than half as many times as two other trackers that do not model shape, including a template-registration method and a single-template particle filter that uses 540 particles.

Particles	Failures	Mean Translation Error	Mean Rotation Error	Mean Time/Frame
100	53	4.63 pixels	0.26 rad	0.97 s
200	45	4.27 pixels	0.20 rad	1.60 s
400	50	4.01 pixels	0.19 rad	2.98 s
800	39	3.68 pixels	0.18 rad	5.64 s
1600	51	3.42 pixels	0.13 rad	10.74 s

Table 2: Incorporating a parts-constellation model into a traditional particle filter without Rao-Blackwellization requires many more samples and still does not achieve the performance level of the RBPF.

807 frames of hand-labeled bee poses. The shape and appearance models were learned simultaneously by applying EM to a subset of 50 frames of training data. The motion model was learned from the incremental translations and rotations between successive frames of the entire training set.

We tested our tracker on a video sequence (810 frames at 720x480 pixels) for which we have hand-labeled ground truth data. All tests were run on a 2.8 GHz Pentium 4 processor. We say a failure occurred when the tracked position differs from the ground truth position by more than half the bee’s width (15 pixels). For these experiments, when the tracker fails, it is reinitialized at the ground truth position for the current frame and resumes tracking.

First, we compared our parts-based RBPF tracker against two other tracking methods (see Table 1). The first tracker (which avoids particles completely) uses an iterative image-alignment method based on Lucas-Kanade. A single bee template is aligned to the image in each frame, starting from the aligned location of the previous frame. The second tracker is a traditional particle filter with a single-template appearance model in which no optimization occurs. This particle filter samples over pose, but shape is not modeled at all. Using a parts-constellation model decreased the number of tracking failures by a factor of 2, from 50 to 24.

For comparison, we show the performance in Table 2 of a parts-based model which does not treat shape analytically and instead samples over both shape and pose. Even after repeatedly doubling the number of particles to 1600, the tracking performance does not improve much beyond the results of the non-shape-based particle filter in Table 1. Because joint samples on shape and pose live in a 12-dimensional space, an extremely large number of particles (and processing time) would be needed to match the per-

formance of our 80-particle RBPF tracker. Only with Rao-Blackwellization do parts-constellation models become efficient enough to be of use in a particle filter.

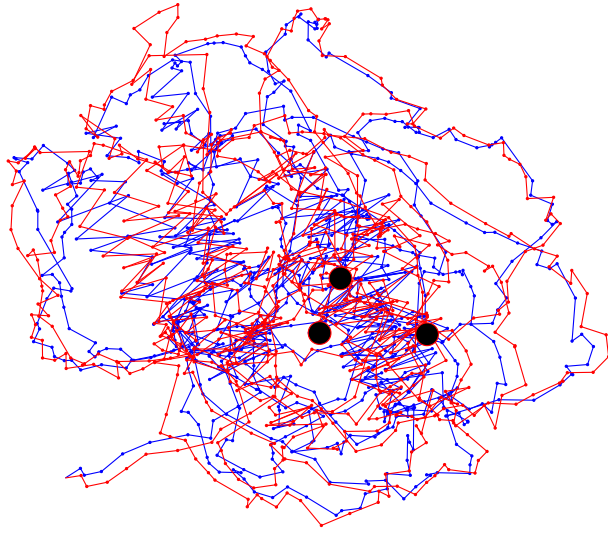
In further experiments, we recorded only 3 failures for an 80-particle RBPF using a more computationally expensive per-pixel background model (see Figure 4), while a 500-particle single-part PF using this model fails roughly twice as often. The remaining failures occurred during the “waggle” portion of the dance, suggesting that a more sophisticated motion model is necessary to further reduce the number of tracker failures.

Note that in theory, we should jointly align all parts of the model simultaneously with respect to both the joint shape distribution and the individual part appearance models. In our implementation, we initialize the shape at the mean of the joint shape distribution, but we are optimizing the appearance of each part individually. Because the appearance model often dominates the shape model, this is a reasonable approximation which is supported by the experiments.

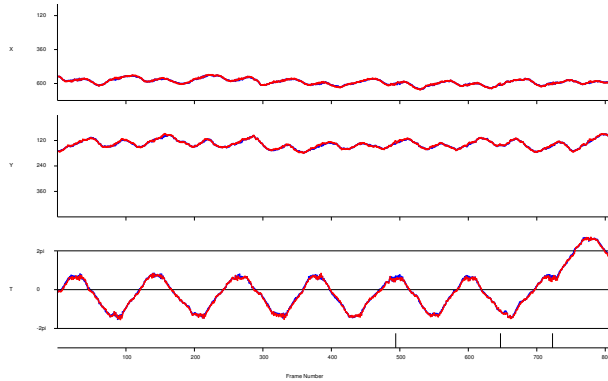
7. Conclusion

The parts-based RBPF tracker presented here reduced tracker failures by a factor of 2. We used somewhat naive appearance and motion models, in part, so that we could isolate and observe more clearly the specific advantages of a parts-based model for difficult tracking tasks. Only with the addition of more sophisticated appearance models (e.g. subspace models) and motion models (e.g. switching linear dynamic systems) would we expect the tracker to perform perfectly. What we have demonstrated is that

- parts-constellation models can be beneficial for some tracking tasks, and that



(a) x vs. y



(b) From top to bottom: x,y,theta vs. time

Tracker	Particles	Failures	Trans. Err.	Rot. Err.	Time/Frame
1-Part PF	500	5	2.29 pixels	0.08 rad	2.01 s
3-Part RBPF	80	3	3.02 pixels	0.13 rad	2.03 s

(c) Tracker performance with a more expensive per-pixel background model.

Figure 4: A parts-based RBPF with 80 particles recorded 3 failures over the course of a bee dance (c). We plot the tracker's performance against ground truth (a) in a 2D view and (b) in a time series view. The ground truth trajectory is plotted in blue, the trajectory returned by the tracker is plotted in red, and failures are indicated with black dots and tick marks in (a) and (b) respectively. Observe that all of the remaining failures occur during the “waggle” portion of the bee dance.

- Rao-Blackwellization enables the efficient use of parts-based models for particle filtering.

References

- [1] S. Baker, F. Dellaert, and I. Matthews. Aligning images incrementally backwards. Technical Report CMU-RI-TR-01-03, CMU Robotics Institute, Pittsburgh, PA, Feb. 2001.
- [2] M.C. Burl, M. Weber, and P. Perona. A probabilistic approach to object recognition using local photometry and global geometry. *Lecture Notes in CS*, 1407:628, 1998.
- [3] J. Carpenter, P. Clifford, and P. Fernhead. An improved particle filter for non-linear problems. Technical report, Department of Statistics, University of Oxford, 1997.
- [4] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.
- [5] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2003.
- [6] N. Freitas. Rao-Blackwellised particle filtering for fault diagnosis. *IEEE Trans. Aerosp.*, 2002.
- [7] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F*, 140(2):107–113, 1993.
- [8] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Eur. Conf. on Computer Vision (ECCV)*, pages 343–356, 1996.
- [9] Z. Khan, T. Balch, and F. Dellaert. A Rao-Blackwellized particle filter for EigenTracking. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2004.
- [10] B. D. Lucas and Takeo Kanade. An iterative image registration technique with an application in stereo vision. In *Seventh International Joint Conference on Artificial Intelligence (IJCAI-81)*, pages 674–679, 1981.
- [11] G.J. McLachlan and T. Krishnan. *The EM algorithm and extensions*. Wiley series in probability and statistics. John Wiley & Sons, 1997.
- [12] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *AAAI Nat. Conf. on Artificial Intelligence*, 2002.
- [13] K. Murphy and S. Russell. Rao-Blackwellised particle filtering for dynamic Bayesian networks. In A. Doucet, N. de Freitas, and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, January 2001.
- [14] D. Schulz, D. Fox, and J. Hightower. People tracking with anonymous and ID-sensors using Rao-Blackwellised particle filters. In *Proceedings of IJCAI*, 2003.
- [15] Leonid Sigal, Sidharth Bhatia, Stefan Roth, Michael J. Black, and Michael Isard. Tracking loose-limbed people. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 421–428, 2004.